Dynamic capacity control for manufacturing environments

Tanja Mlinar

IÉSEG School of Management, France

t.mlinar@ieseg.fr

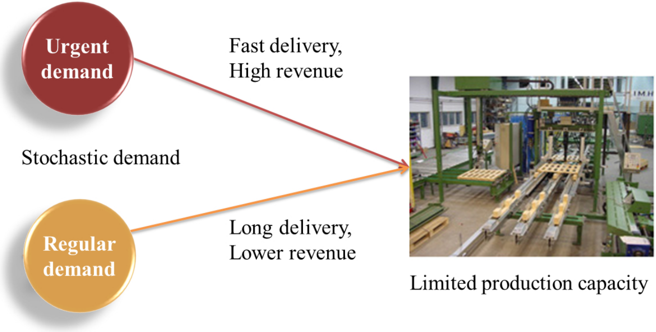
December 03, 2015

Dynamic capacity control

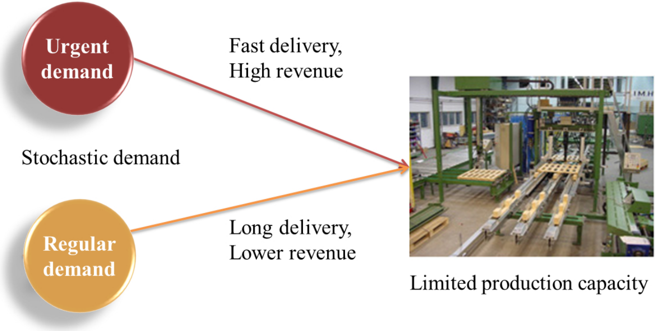
- **Make-To-Order environments**
  - Revenue management for operations with urgent orders, (with P. Chevalier, A. Lamas, L. Lu)
    - *European Journal of Operational Research*
  - Dynamic admission control for two customer classes with stochastic demands and strict due dates, (with P. Chevalier)
    - *International Journal of Production Research*

- **Hybrid Make-To-Stock and Make-To-Order environments**
  - Revenue maximization by integrating order acceptance and stocking policies, (with P. Chevalier, A. Lamas)
    - *To be submitted*

Dynamic capacity control for Make-To-Order environments



Dynamic capacity control for Make-To-Order environments



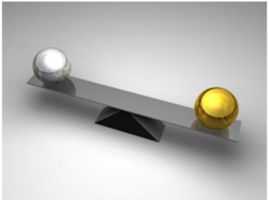
The challenge for the company

To make decisions of accepting or rejecting customer requests in order to maximize its profit and to fullfill the promises agreed with the customers.

Motivation

A trade-off between the amount of low profitable orders to accept and the available capacity to allocate future high profitable orders.

- Accepting too many regular orders lead to loosing the possibility to serve more profitable orders.
- Accepting too few regular orders lead to under-utilization of capacity.



Examples

Iron-Steel Industry, Heating, Ventilation and Air-Conditioning (HVAC), High Fashion Clothing...

The objective of this work

To provide an approach which maximizes the expected net profit of the company by selectively accepting orders from two different demand streams.

Methodology

- We study the dynamic capacity allocation problem under demand uncertainty in arrivals and order sizes.
- An optimal order acceptance policy (an MDP formulation).
  - The system state space explodes quickly for larger instances.

Methodology

- We study the dynamic capacity allocation problem under demand uncertainty in arrivals and order sizes.
- An optimal order acceptance policy (an MDP formulation).
    - The system state space explodes quickly for larger instances.
- A threshold-based heuristic policy
    - To reduce the cardinality of possible policies, and thus the computational requirements.

Methodology

- We study the dynamic capacity allocation problem under demand uncertainty in arrivals and order sizes.
- An optimal order acceptance policy (an MDP formulation).
    - The system state space explodes quickly for larger instances.
- A threshold-based heuristic policy
    - To reduce the cardinality of possible policies, and thus the computational requirements.
- We evaluate whether the solutions are robust to changes:
    - in operational conditions
    - in actual demand from its estimation (forecast errors).

Problem Description

Assumptions and Parameters

- Infinite-discrete planning horizon.
- Unit discrete capacity.
- Two customer classes ($k = \{1, 2\}$):
  - Stochastic demand:
    - $D_k = B_{k,s}$ with arrival probability p_k and probability of occurrence of size s $q_{k,s}$, and $D_k = 0$ with probability $1 - p_k$.
    - Number of order sizes of class k, n_k .
  - Revenue per unit capacity r_k: $r_1 > r_2$.
  - Lead time l_k: $L_1 < L_2$.

Problem Description

Assumptions and Parameters

- When an order is placed, processing times are known with certainty.
- No changeovers.
- Accept all or nothing of an order.
- The regular orders can be selectively rejected, urgent orders can only be passively rejected.
- Shipping on or before the due date.

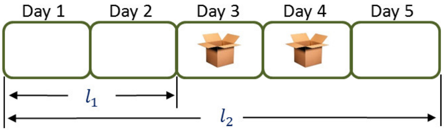
Problem Description: Sequence of Events

0. Current state

Problem Description: Sequence of Events

0. Current state

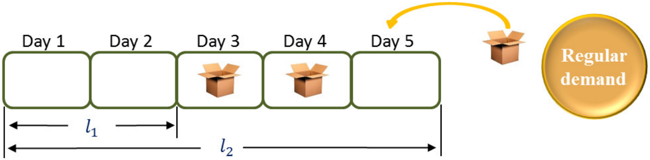
1. Arrival of class 2



Problem Description: Sequence of Events

0. Current state

1. Arrival of class 2

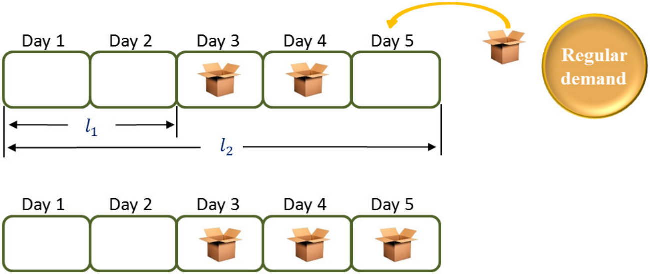


Problem Description: Sequence of Events

0. Current state

1. Arrival of class 2

2. Accept class 2



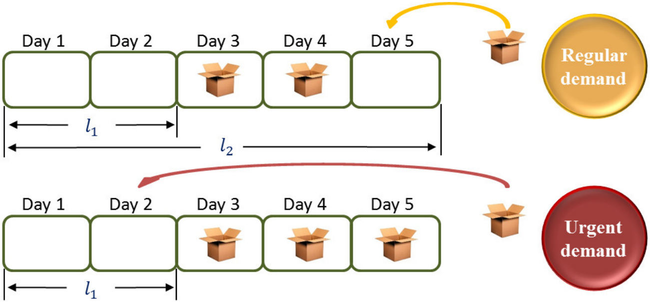
Problem Description: Sequence of Events

0. Current state

1. Arrival of class 2

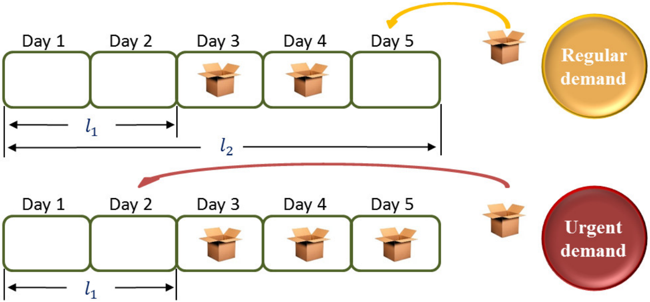
2. Accept class 2

3. Arrival of class 1



Problem Description: Sequence of Events

0. Current state
1. Arrival of class 2

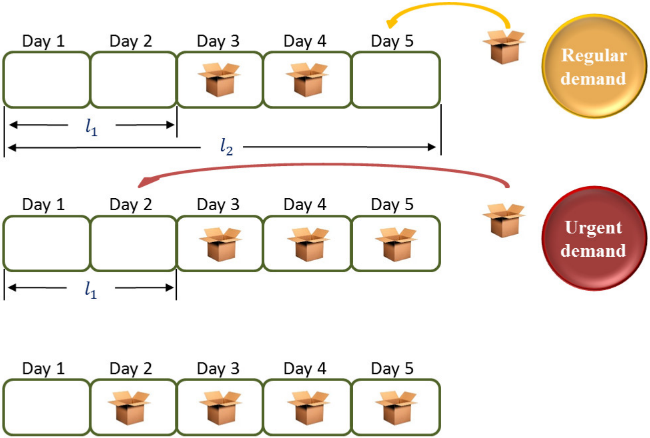


2. Accept class 2
3. **Arrival of class 1**

Problem Description: Sequence of Events

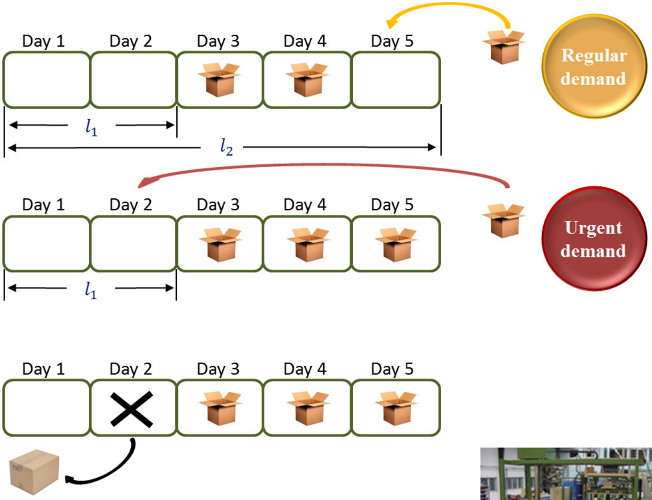
0. Current state
1. Arrival of class 2
2. Accept class 2
3. Arrival of class 1

4. Accept class 1



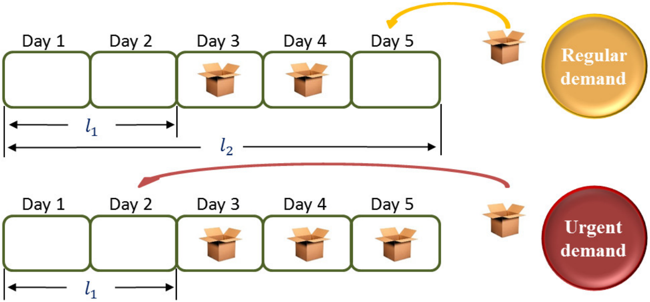
Problem Description: Sequence of Events

0. Current state
1. Arrival of class 2
2. Accept class 2
3. Arrival of class 1
4. Accept class 1
5. **Production**



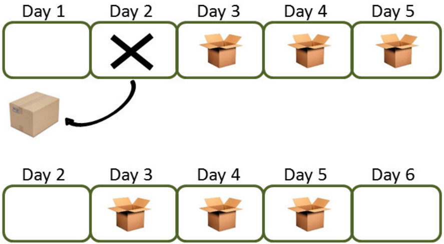
Problem Description: Sequence of Events

0. Current state
1. Arrival of class 2



2. Accept class 2
3. Arrival of class 1

4. Accept class 1
5. Production



6. Update to the next period

Markov Decision Process Formulation

- *Three reservation vectors* are defined:
  - vector $\mathbf{x}$ represents the system state at the beginning of each period
    - $\mathbf{x}[0] \in \{0, 1, \ldots, L_1\}$ the total reserved capacity till L_1 the period.
    - $\mathbf{x}[\mathbf{j}] \in \{0, 1\}$, the reserved capacity of $L_1 + j$ period, where $j = 1, 2, \cdots, L_2 - L_1$.
  - vector $\tilde{\mathbf{x}}$ represents the system state upon the acceptance of regular order.
  - vector $\hat{\mathbf{x}}$ represents the system state upon the update to the next period.

Markov Decision Process Formulation

- *Three reservation vectors* are defined:
  - vector $\mathbf{x}$ represents the system state at the beginning of each period
    - $\mathbf{x}[0] \in \{0, 1, \ldots, L_1\}$ the total reserved capacity till L_1 the period.
    - $\mathbf{x}[\mathbf{j}] \in \{0, 1\}$, the reserved capacity of $L_1 + j$ period, where $j = 1, 2, \cdots, L_2 - L_1$.
  - vector $\tilde{\mathbf{x}}$ represents the system state upon the acceptance of regular order.
  - vector $\hat{\mathbf{x}}$ represents the system state upon the update to the next period.
- *Acceptance policy:* vector $\mathbf{a}$, number of elements n_2.

Optimal order acceptance policy

3 order sizes of class 2:

Feasible policies:

Optimal order acceptance policy

Markov Decision Process Formulation

- *System state space:* $s = (L_1 + 1) \cdot 2^{L_2 - L_1 - 1}$.
  - *Reservation vector* $\mathbf{x}$
    - $\mathbf{x}[0] \in \{0, 1, \ldots, L_1\}$
    - $\mathbf{x}[\mathbf{j}] \in \{0, 1\}$, where $j = 1, 2, \cdots, L_2 - L_1$.

Markov Decision Process Formulation

- *System state space:* $s = (L_1 + 1) \cdot 2^{L_2 - L_1 - 1}$.
    - *Reservation vector* $\mathbf{x}$
        - $\mathbf{x}[0] \in \{0, 1, \ldots, L_1\}$
        - $\mathbf{x}[\mathbf{j}] \in \{0, 1\}$, where $j = 1, 2, \cdots, L_2 - L_1$.
- *Action space:* $2^{n_2} \cdot (L_1 + 1) \cdot 2^{L_2 - L_1 - 1}$.
    - *Acceptance policy* $\mathbf{a}$: number of elements n_2.

Optimal order acceptance policy

Linear Programming model is developed to solve the MDP problem

$$\begin{aligned} & \max g \\ s.t. \qquad & V(\mathbf{x}) + g \leq \mathbb{E}_{\mathbf{D}} \left\{ R(\mathbf{x}, \mathbf{D}, \mathbf{a}) + V\left(\widehat{\mathbf{x}}(\mathbf{x}, \mathbf{D}, \mathbf{a})\right) \right\}, \forall \mathbf{x}, \forall \mathbf{a}. \end{aligned}$$

- Number of constraints: $2^{n_2} \cdot (L_1 + 1) \cdot 2^{L_2 - L_1 - 1}$

Optimal order acceptance policy

Limitations:

- Obtaining the optimal policy becomes hard when $L_2 - L_1$ and n_2 increase.
  - For example, when $L_1 = 5$, $L_2 = 25$ and $n_2 = 5$:
    - The action space: $100,663,000$.
    - The system state space: $3,146,000$.

Optimal order acceptance policy

Limitations:

- Obtaining the optimal policy becomes hard when $L_2 - L_1$ and n_2 increase.
  - For example, when $L_1 = 5$, $L_2 = 25$ and $n_2 = 5$:
    - The action space: $100,663,000$.
    - The system state space: $3,146,000$.

Our objective:

- To provide efficient heuristic policies by reducing the complexity of the formulation.
  - To reduce the action space.
  - To reduce the system state space.

Threshold-based order acceptance policy

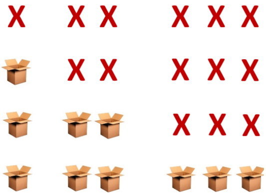
- **To reduce the action space:**
  - We propose a threshold based policy $t = \{0, 1, 2, \cdots n_2\}$.
    - We modify the MDP formulation in order to provide such a policy.
  - New action space: $(n_2 + 1) \cdot (L_1 + 1) \cdot 2^{L_2 - L_1 + 1}$.

Threshold based order acceptance policy

3 order sizes of class 2:

Feasible policies:

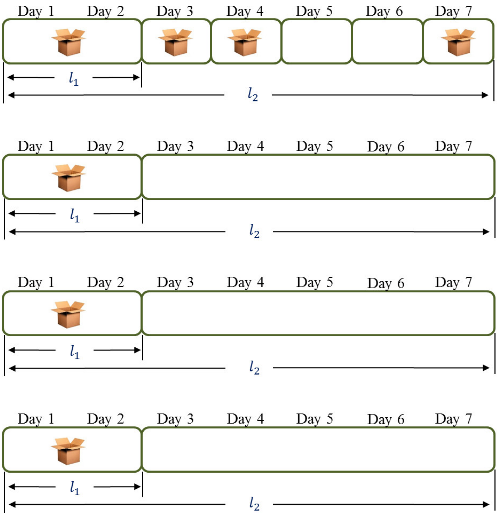
Threshold policies:



Threshold-based aggregation heuristics

- **To reduce the system state space:**
  - We aggregate the distributional information in the interval $L_2 - L_1 - 1$.
    - The reduced formulation is used to generate heuristic policies.

Threshold-based Full Aggregation Heuristic T-FAH



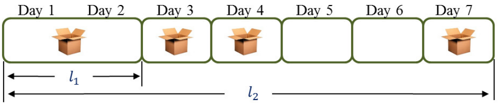
MDP Formulation
x=(1, 1, 1, 0, 0, 1)

FAH Optimistic Scenario
x_f=(1, 3)

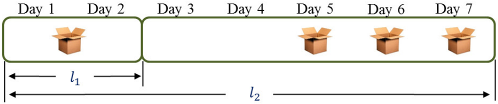
FAH Pessimistic Scenario
x_f=(1, 3)

FAH Realistic Scenario
x_f=(1, 3)

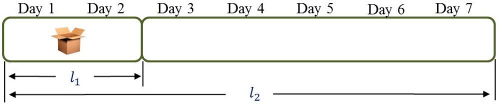
Threshold-based Full Aggregation Heuristic T-FAH



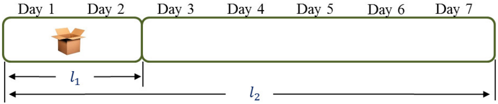
MDP Formulation
x=(1, 1, 1, 0, 0, 1)



FAH Optimistic Scenario
x_f=(1, 3)

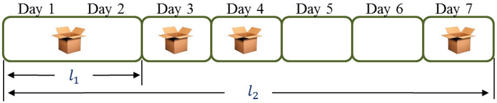


FAH Pessimistic Scenario
x_f=(1, 3)



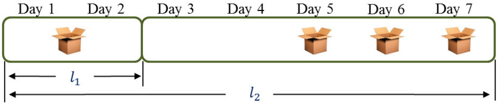
FAH Realistic Scenario
x_f=(1, 3)

Threshold-based Full Aggregation Heuristic T-FAH



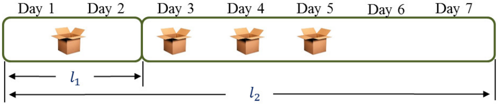
MDP Formulation

x=(1, 1, 1, 0, 0, 1)



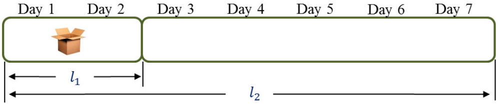
FAH Optimistic Scenario

x_f=(1, 3)



FAH Pessimistic Scenario

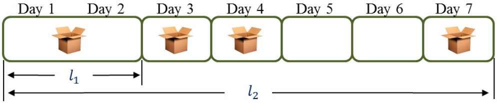
x_f=(1, 3)



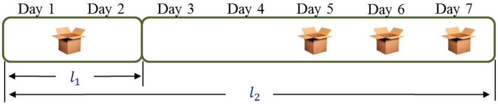
FAH Realistic Scenario

x_f=(1, 3)

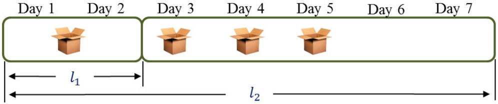
Threshold-based Full Aggregation Heuristic T-FAH



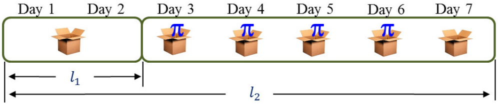
MDP Formulation
x=(1, 1, 1, 0, 0, 1)



FAH Optimistic Scenario
x_f=(1, 3)

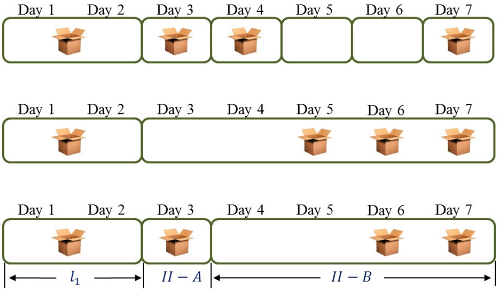


FAH Pessimistic Scenario
x_f=(1, 3)



FAH Realistic Scenario
x_f=(1, 3)

- Parameter $z \in \{0, ..., L_2 - L_1 - 1\}$ controls the level of aggregation in each state of the system
- Size of the state space $(L_1 + 1) \cdot (L_2 - L_1 + 1 - z) \cdot 2^z$



MDP Formulation

x=(1, 1, 1, 0, 0, 1)

size $(L_1+1) \cdot 2^{L_2-L_1+1}$

FAH Optimistic Scenario

x_f=(1, 3)

size $(L_1+1) \cdot (L_2-L_1+1)$

PAH Optimistic Scenario

x_f=(1,1,2), z=1

size $(L_1+1) \cdot (L_2-L_1+1-z) \cdot 2^z$

Numerical study

Reduction in the computational time

L_1, L_2	Policy	n_1, n_2	MDP	Aggregation Heuristics	
				FAH	PAH (z=4)
6, 18	threshold	3, 7	8.2	0.02	0.59
	unconstrained		133.8	0.30	6.18
6, 30	threshold	3, 7	*	0.08	8.17
	unconstrained		*	0.74	38.77

Note: Values are in seconds.

Numerical study

Reduction in the computational time

L_1, L_2	Policy	n_1, n_2	MDP	Aggregation Heuristics	
				FAH	PAH (z=4)
6, 18	threshold	3, 7	8.2	0.02	0.59
	unconstrained		133.8	0.30	6.18
6, 30	threshold	3, 7	*	0.08	8.17
	unconstrained		*	0.74	38.77

Note: Values are in seconds.

Reduction in the dimension of the admission problem

L_2	Policy	n_2	MDP	Aggregation Heuristics	
				FAH	PAH (z=4)
30	threshold	7	469,762	1	65
	unconstrained		7,516,192	21	1,032

Note: values in 10^3.

Structure of the optimal policy

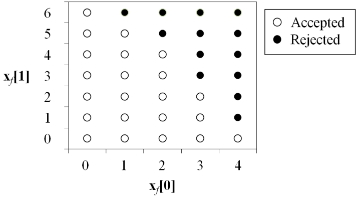
- The threshold policy is the optimal policy for majority of instances.
  - The largest optimality gap is 0.133%.
  - For a very small number of states there is a combinatorial effect related to order sizes.

Structure of the optimal policy

- The threshold policy is the optimal policy for majority of instances.
  - The largest optimality gap is 0.133%.
  - For a very small number of states there is a combinatorial effect related to order sizes.

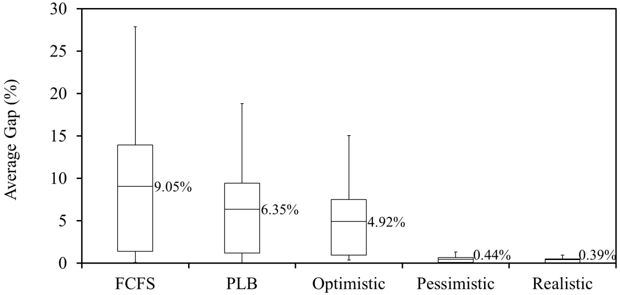
Structure of the FAH heuristic policy

- Corresponds to a threshold in the values of $\mathbf{x_f}[\mathbf{0}]$ and $\mathbf{x_f}[\mathbf{1}]$.



Robustness to Changes in Operational Parameters

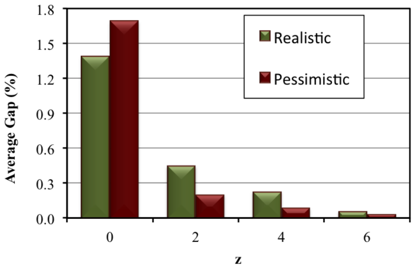
- T-FAH under the realistic and pessimistic scenarios is superior over the other methods.



- The largest optimality gaps under the pessimistic and realistic scenario are lower than the average optimality gap under the PLB policy obtained by a myopic approach (Barut and Sridharan, 2005).

Numerical study

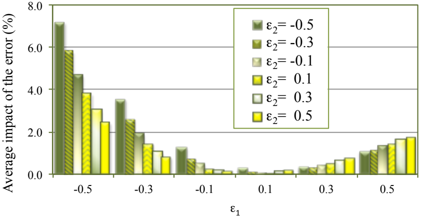
Efficiency of the T-PAH for the worst instances



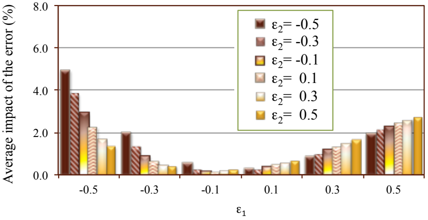
- The T-PAH shrinks the optimality gap for the worst 10% of instances.

Robustness to Forecast Errors

T-FAH Realistic Scenario



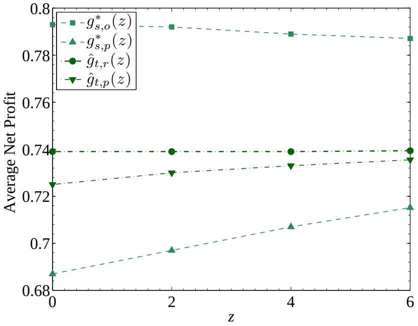
T-FAH Pesimistic Scenario



Observations

- The impact of errors is considerably smaller when $\epsilon_1 > 0$.
- The pessimistic scenario underestimates the available capacity in Interval I, the average performance degradation is larger when $\epsilon_1 > 0$

Efficiency of the threshold based policy for large instances



- The average optimality gap of the T-FAH under the realistic scenario is lower than 6.06%.

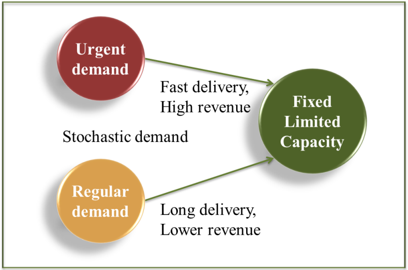
Conclusions

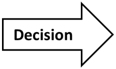
- The work represents a first work which deals with order acceptance decisions with heterogeneous customer classes by explicitly taking into account uncertainty in demand and a rolling planning horizon
- We showed that the optimal policy is a threshold policy for most of the instances.
- Our threshold heuristic policies are near optimal and can be obtained very quickly.
- Solutions are robust to changes in operational parameters and forecast errors.

Conclusions

- The work represents a first work which deals with order acceptance decisions with heterogeneous customer classes by explicitly taking into account uncertainty in demand and a rolling planning horizon
- We showed that the optimal policy is a threshold policy for most of the instances.
- Our threshold heuristic policies are near optimal and can be obtained very quickly.
- Solutions are robust to changes in operational parameters and forecast errors.
- Future research
  - To determine under which conditions the optimal policy has a threshold structure.
  - To prove analytically that the heuristic policy is a threshold in values of $x_f[0]$ and $x_f[1]$.

Dynamic Capacity Control:
Hybrid Make-to-Order and Make-to-Stock Environments





Assumptions:

- Unit discrete capacity.
- Deterministic processing times.
- Accept all or nothing of an incoming order.
- No tardiness.
- **Storage capacity of I_{max} units.**
- **Unit holding cost per period.**

Problem Description

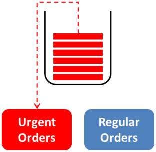
In order to maximize its profit, the manufacturer must decide:

- whether to accept or reject a regular order.
- whether to increase or not the inventory level.

Problem Description

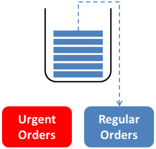
Four types of stock problems

Stock of urgent units



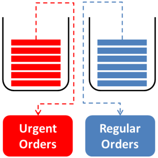
Ready to pay holding costs to meet short lead times.

Stock of regular units



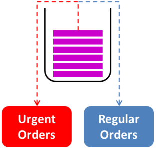
Regular orders correspond to more standardized products.

Stock of urgent and regular units



Heterogeneous products.

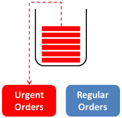
Joint stock of units



Products only differ in lead times.

Methodology: Decisions

Stock of urgent units



$$A(x,s) = \begin{cases} A1 & : \text{raise inventory level and accept regular order.} \\ A2 & : \text{raise inventory level and reject regular order.} \\ A3 & : \text{do not raise inventory level and accept regular order.} \\ A4 & : \text{do not raise inventory level and reject regular order.} \end{cases}$$

Methodology

We formulate the problem as a Markov Decision Process (MDP)

- The state of the system is described by (x, s)
  - x keeps track of capacity that has been reserved for processing;

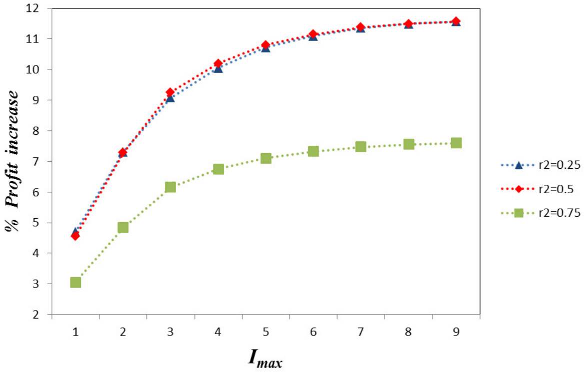
$$x[j] = \begin{cases} 1 & \text{if the capacity of } j\text{th period is reserved} \\ 0 & \text{otherwise} \end{cases}, \text{for } j = 1..., L_2.$$

  - s denotes the inventory level of class 1, i.e., $s \in \{0, ..., Imax\}$.
- $A(x, s)$ represents the decision made upon the arrivals of class 1 and class 2 orders,
- We maximize the expected long run revenues.

$$V(\mathbf{x}, s) + g = \mathbb{E}_{\mathbf{D}} \left\{ \max_{A} \left\{ R(\mathbf{x}, s, \mathbf{D}, A) + V\left(\mathbf{x}^{(\mathbf{v})}(\mathbf{x}, s, \mathbf{D}, A), s^{(ii)}(\mathbf{x}, s, \mathbf{D}, A) \right) \right\} \right\}, \forall \mathbf{x}, \forall s,$$

Preliminary Results

Stock of urgent units



Conclusions and Future Research

We studied the order acceptance and inventory problem for MTO/MTS environments.

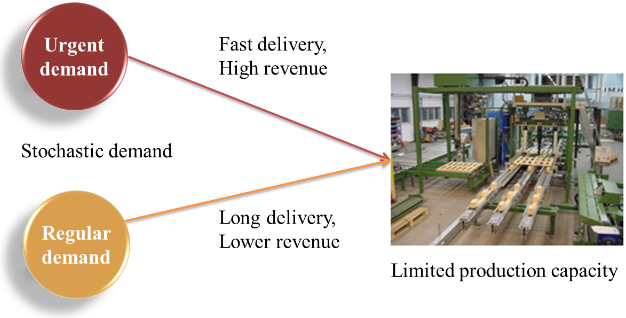
The proposed MDP formulation considers different types of inventory problems.

- The resolution implies high computational requirements.

Future research

- To propose an heuristic approach consisting in a parametric aggregation of the state space.

Dynamic capacity control for manufacturing environments



Thank you for your attention!